A new clustering algorithm based on the chemical recognition system of ants

Nicolas Labroche, Nicolas Monmarché, Gilles Venturini

A new clustering algorithm based on the chemical recognition system of ants

Nicolas Labroche, Nicolas Monmarché, Gilles Venturini
Laboratoire d'Informatique de l'Université de Tours, EA 6300
email: labroche,monmarche,venturini@univ-tours.fr

Abstract

In this paper, we introduce a new method to solve the unsupervised clustering problem, based on a modelling of the chemical recognition system of ants. This system allow ants to discriminate between nestmates and intruders, and thus to create homogeneous groups of individuals sharing a similar odor by continuously exchanging chemical cues. This phenomenon, known as "colonial closure", inspired us into developing a new clustering algorithm and then comparing it to a well-known method such as K-MEANS method. Our results show that our algorithm performs better than K-MEANS over artificial and real data sets, and furthermore our approach requires less initial information (such as number of classes, shape of classes, limitation in the types of attributes handled).

1 Introduction

The efficiency of real ants collective behaviors has led number of computer scientists to create and propose novel and successful approaches to problem solving. For instance, modelling collective behaviors has been used in the well known algorithmic approach Ant Colony Optimization (ACO)([4]) in which pheromone trails are used. In the same way, other ants-based clustering algorithms have been proposed ([12], [8], [13]). In these studies, researchers have modelled real ants abilities to sort objects. Artificial ants may carry one or more objects and may drop them according to given probabilities. These agents do not communicate directly with each others, but they may influence each others through the configuration of objects on the floor. Thus, after a while, these artificials ants are able to construct groups of similar objects, a problem which is known as data clustering.

In this paper, we focus on another important real ants collective behavior, namely the construction of a colonial odor and its use for determining the ants

nest membership. As far as we know, this model has not yet been applied to any task in problem solving, and we show here how it can be used in data clustering.

The remaining of this article is organized as follows: section 2 describes the main principles of the real ants recognition system. Section 3 presents the first data clustering algorithm that uses this new model: ANTCLUST. Section 4 details experimental tests on benchmarks and their comparisons with standard approaches. Finally, section 5 concludes on future extensions of this new model.

2 Main properties of real ants recognition system

Every day, real ants have to solve a crucial recognition problem when they meet: they have to decide whether they belong to the same nest or not, in order to guaranty the survival of the nest. This phenomenon is known as "colonial closure". It mainly relies on continuous exchanges and updates of chemical cues on the ants cuticle and in their post-pharyngeal gland (PPG), determining, as an identity card, their belong to the nest. Thus, each ant has its own view of its colony odor at a given time, and updates it continuously. By this way, an ant preserves its nest from being attacked by predators or parasites and reinforces its integration in nest [1][11]. We are going to briefly developp the main principles of such a recognition system from the generation of ants odor to its evolution and its sharing. More details are given in [9] with a complete related mathematical model that is very close to the biological reality.

2.1 Principles of the recognition system

In ants societies, according to [5] [6], nestmate recognition implies a complex system allowing discrimination between individuals, based on four distinct levels of analysis for each ant:

1. The existence of an individual chemical odor (or "Label"), partially constructed by each ant, species and environment dependant, spread over its cuticle.
2. A mecanism of chemical reception allowing the reading of the Labels of encountered ants
3. A model of reference (or "Template"), that indicates what type of odor nestmates should have on their cuticle. The Template is learned and then continuously updated.
4. A set of decision rules that triggers behaviors according to the similarity between the Template and the perceived Label of an encountered ant.

According to [6], the recognition between two ants relies on the detection of phenotypic differences ("phenotype matching"). Thus, each ant compares the other's Label to its reference model ("Template") to resolve the recognition problem.

Labels are mainly made of hydrocarbons that ants are able to synthetise and chemical substances that ants may extract from their food or their environment. Hydrocarbons may vary qualitatively and quantitatively according to species.

Moreover, many chemical substances can modify the cuticular odor and hence influence inter-individuals recognition process. We cite among these: the colony's queen [10], [6], [2], the food [6], the environment of the nest [6].

2.2 Odor ontogenesis and evolution

At the early stage of their life, young ants, when fed by other colony members, physically impregnate nestmates' odors and learn them as a first Template. Their Label are then only defined by their own genetic information. After a short time, ants are able to synthetise their own hydrocarbons and thus can reinforce their Label by spreading their PPG's content to their cuticle ("individual licking"). The homogeneous sharing of all nestmates odors in a colony is achieved by trophallaxies (an ant decants its PPG contents in an other's PPG), by "social licking" (each ant spread a portion of its PPG over the other's cuticle) or by simple contacts (only cuticular substances are exchanged).

3 Clustering algorithm ANTCLUST

3.1 The clustering problem

In this paper, we focus on the unsupervised clustering problem in which the goal is to find groups of similar objects as close as possible to the natural partition of the given data set. No assumptions are made about the representation of the objects. They may be described with numerical or symbolic values or with first order logic. All we need here is the definition of a similarity measure which takes a couple of objects i and j as input and outputs a value $Sim(i, j)$ between 0 and 1. Value 0 means that the two objects are totally different, 1 means that they are identical.

3.2 Main principles of ANTCLUST algorithm

The main idea in this new model is as follows: one object is assigned to each artificial ant and represents the genetic part of the ant's odor. We detail hereafter how its Label and its Template are represented (see 1).

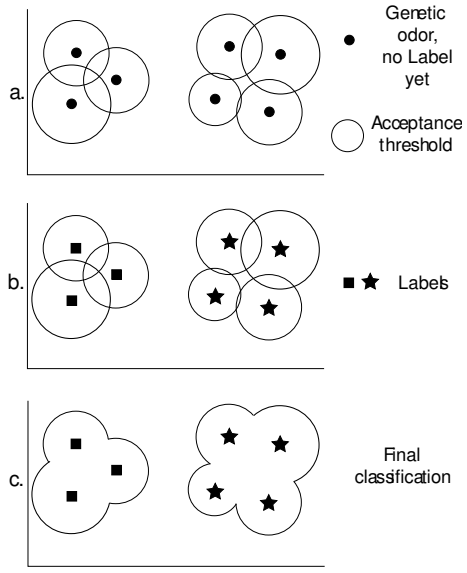

Figure 1: Principles of ANTCLUST. Labels and Templates are represented in a 2D-space for a better understanding. In a. ants have no Label and are just described by their Genetic odor. In b. the first labels have been computed by the algorithm. In c. the final classification groups in the same nest, the ants that share a similar Label.

For one ant i, we define the following parameters:

- The Label $Label_i$ is determined by the belonging nest of the ant i and is simply coded by a number, representative of the nest. At the beginning, ants are not under the influence of any nest, so $Label_i$ equals 0. This Label will evolve over time until each ant has found its best nest.

- The Template is defined half by the genetic odor $Genetic_i$ of the ant and half by an acceptance threshold $Template_i$. The first one corresponds to an object of the data set and can not evolve during the algorithm. The latter is learned during an initialization phase, similar to the real ants ontogenesis period, in which each artificial ant will meet others and each time will evaluate the genetic odors similarities. The resulting Template threshold $Template_i$ is a function of all the similarities observed during this period. This acceptance threshold $Template_i$ is dynamic and is updated after every encounters realised by ant i.

- An estimator M_i that reflects if the ant i is successful during its meetings with all encountered ants or not. Since a young ant has not realised any meeting, $M_i = 0$ at time $t = 0$. M_i estimates the size of the nest to which i belongs to (i.e. ants with the same $Label$). M_i is simply increased when ant i meets an other ant with the same Label and decreased when Labels are different (see section 3.4.1).

- An estimator M_i^+ which measures how well accepted is ant i in its nest. It is increased when ant i meets another ant with the same Label and when both ants accept each other and decreased when there is no acceptance between ants (see section 3.4.1).

- An age A_i which, at the beginning, equals 0 and is used when updating acceptance threshold.

- Estimates of the maximal similarity $Max(Sim(i, \cdot))$ and mean similarity $\overline{Sim}(i, \cdot)$ observed during its meeting with other ants.

We simulate meetings between ants by randomly selecting two ants at each iteration. Label and acceptance threshold are changed according to behavioral rules (see section 3.4.1). At the end of the execution of the algorithm, the gathering of ants in a finite number of nest where nestmates are more similar to each other than the ants of other colonies, provides a partition of the set of objects, which was the goal to be reached.
The detail of ANTCLUST main algorithm is written hereafter.

Algorithm 1: ANTCLUST main algorithm
ANTCLUST()

(1) Initialize the ants:
(2) $Genetic_i \leftarrow i^{th}$ objects of the data set
(3) $Label_i \leftarrow 0$
(4) $Template_i$ is initialized (see section 3.3)
(5) $M_i \leftarrow 0$, $M_i^+ \leftarrow 0$, $A_i \leftarrow 0$
(6) Simulate Nb_{ITER} iterations during which two ants, that are randomly chosen, meet
(7) Delete nests with less than $P \times n$ ($P \ll 1$) ants
(8) Re-assign each ant having no more nest to the nest of the most similar ant found that have a nest.

We now detail two fundamental mechanisms of our method: initialization of young ants and the set of behavior rules underlying the meeting process.

3.3 Initialization of young ants

We have copied the creation phase of young artificial ants from the ontogenesis period of biological ants, during which they learn a template of their colony odor, that will allow them to accept or reject encountered ants.

Thus, we consider that the template can be defined as an acceptance threshold $Template_i$, for each ant i, that will be learned during a given number of random meetings. At the end of this period, the ant i possesses values of mean and maximal similarities, which are used to define the Template at the beginning as shown in the following equation (1).

$$Template_i \leftarrow \frac{\overline{Sim}(i,\cdot) + Max(Sim(i,\cdot))}{2} \tag{1}$$

During its meetings, the ant i will progressively make its mean and maximal similarities values evolve, in order to continuously update its Template threshold $Template_i$ according to equation (1).

3.4 Ants meetings resolution

The crucial point of our method concerns the resolution of meetings. It allows ants to share a common Label with individuals with compatible templates. We consider thereafter two ants i and j. We define that there is acceptance (or recognition) between i and j (see figure 2):

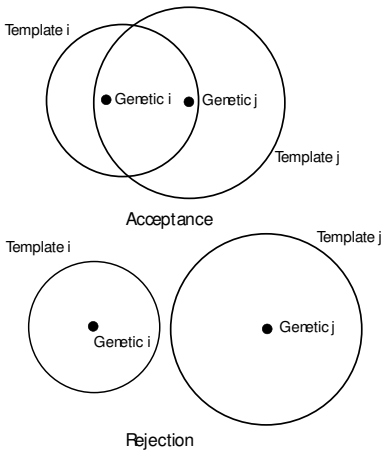

Figure 2: Principle of acceptance and rejection between two ants i and j

$$\begin{aligned} Acceptance(i,j) \Leftrightarrow & (Sim(i,j) > Template_i) \\ & \wedge (Sim(i,j) > Template_j) \end{aligned} \tag{2}$$

3.4.1 Behavioral rules associated with meetings

R_1 New nest creation :
If $(Label_i = Label_j = 0)$ and Acceptance(i,j) Then Create a new Label $Label_{NEW}$ and $Label_i \leftarrow Label_{NEW}, Label_j \leftarrow Label_{NEW}$. If $Acceptance$ is false then rule R_6 is applied.

R_2 Adding an ant with no Label to an existing nest:
If $(Label_i = 0 \wedge Label_j \neq 0)$ and Acceptance(i,j) Then $Label_i \leftarrow Label_j$. The case $(Label_j = 0 \wedge Label_i \neq 0)$ is handled in a similar way.

R_3 "Positive" meeting between two nestmates:
If $(Label_i = Label_j) \wedge (Label_i \neq 0) \wedge (Label_j \neq 0)$ and Acceptance(i,j) Then Increase M_i, M_j, M_i^+ and M_j^+.

By increasing (3) or decreasing (4) a variable x we respectively mean:

$$x \leftarrow (1-\alpha) \times x + \alpha \tag{3}$$

$$x \leftarrow (1-\alpha) \times x \tag{4}$$

(Here we choose $\alpha = 0.2$, because this is useful to "track" those evolving quantities)

R_4 "Negative" meeting between two nestmates:
If $(Label_i = Label_j) \wedge (Label_i \neq 0) \wedge (Label_j \neq 0)$ and Acceptance(i,j)= $False$ Then Increase M_i, M_j and Decrease M_i^+ and M_j^+. The ant x (x=i, x=j) which possesses the worst integration in the nest ($x|M_x^+ = Min_{k\in[i,j]}M_k$) loses its Label and thus has no more nest ($Label_x \leftarrow 0$, $M_x \leftarrow 0$ and $M_x^+ \leftarrow 0$).

R_5 Meeting between two ants of different nests:
If $(Label_i \neq Label_j)$ and Acceptance(i,j) Then Decrease M_i and M_j. The ant x with the lowest M_x (i.e. ant belonging to the smallest nest) changes its nest and belongs now to the nest of the encountered ant.

R_6 Default rule: If no other rule applies then nothing happens.

3.4.2 Analysis of behavioral rules

This section aims at briefly describing the rules mentioned herebefore.

The rule R_1 has a fundamental role because it is the only creative rule in the method. No other rule can generate a new Label for a new nest. It causes the gathering of similar ants in the very first clusters. The latter will be used as "seeds" to generate the final clusters. According to this rule, a cluster contains at least two objects.

The rule R_2 enlarges the cluster by adding an ant with no nest to a nest in which there exists a similar ant.

The rule R_3 simply increments the estimators M and M^+ in case of acceptance between the two ants.

The rule R_4 permits to remove ants that were accepted when the nest profile was not clearly defined because there were not enough ants in it. By means of this rule, the worst integrated ants in a nest can be rejected and then reset. This allows bad or not optimally clustered objects to change their belonging cluster, which improves the results of the algorithm.

The rule R_5 is also very important because it allows the gathering of similar clusters with one bigger than the other, the small one being absorbed by the other one. In fact, at the beginning, there are lots of clusers and this rule significantly decreases their number by gathering small sub-clusters into one bigger one.

The rule R_6 happens when no other rule applies.

4 Experiments and results

In this section, we compare our method ANTCLUST to the K-MEANS algorithm [7]. The latter is initialized with 10 clusters randomly generated, so we will refer to it as 10-MEANS hereafter. Before detailling the settings of experiments, the benchmarks used for evaluation must be introduced.

4.1 Benchmarks and Experimental settings

In order to test and compare the clustering abilities of the two methods, we use randomly generated and real data sets attributes-based representations. For more detail on the data sets, see [13]. Namely, there are: $\text{ART}_{i,i\in[1,8]}$ as artificial data sets and for real ones: IRIS, GLASS, PIMA, SOYBEAN and THYROID. Concerning artificial data sets, ART1, ART2, ART3, ART5 and ART6 are generated by gaussian laws with different difficulties (irrelevant attributes, clusters overlap), ART4 data set is generated by a uniform law and finally ART7 and ART8 correspond to white noise. The main characteristics of data are summarized in table 1.

All evaluations have been conducted over 50 runs for each data set and each method. Concerning ANTCLUST, each test corresponds to 300000 iterations. During each of these iterations, two randomly chosen ants meet. Results are shown in table 2. The following fields are introduced in the table 1 for each data file: the number of objects (*"#Objects"*) and their associated number of attributes(*"#Attributes"*), the number of clusters expected to be found in the data (*"#Clusters"*).

4.2 Similarity between objects and clustering error evaluation

This section introduces two major definitions to help understand the results that are developped hereafter. The following equations present on the one hand, how we compute similarity and on the other hand, the mathematical expression used to compute the clustering error during our evaluations.

Considering the similarity definition, each object is represented by a set of attributes, each of them having a data type θ_k among the Nb_{Types} existing data types (i.e. numeric, symbolic, ...). Global similarity between two objects o_i and o_j can then be defined:

Datas	#Objects	#Attributes	#Clusters
Art1	400	2	4
Art2	1000	2	2
Art3	1100	2	4
Art4	200	2	2
Art5	900	2	9
Art6	400	8	4
Art7	100	2	1
Art8	1000	2	1
Iris	150	4	3
Glass	214	9	7
Pima	798	8	2
Soybean	47	35	4
Thyroid	215	5	3

Table 1: Main characteristics of the data sets.

$$Sim(o_i, o_j) = \frac{1}{Nb_{Types}} \times \sum_{k=1}^{Nb_{Types}} Sim_{\theta_k}(o_i, o_j) \tag{5}$$

$$Sim_{\theta_k}(o_i, o_j) = 1 - \left(\frac{1}{Occ\,(\theta_k)} \times \sum_{k=1}^{Occ(\theta_k)} \Delta_{\theta_k}(o_i, o_j) \right) \tag{6}$$

where $Sim_{\theta_k}$is the similarity computed between all the attributes of type θ_k for both objects o_i and o_j, $Occ(\theta_k)$ the number of times that data type θ_k is used to describe an object o and finally Δ_{θ_k} a function that computes the dissimilarity between two attributes of the compared objects o_i and o_j having data type θ_k. Description of the Δ_{θ_k} functions won't be too detailled, but we formalize hereafter the Δ_{Num} (7) and Δ_{Symb} (8) respectively, for a pair (i, j) of numeric or symbolic values:

$$\Delta_{Num}(i, j) = \begin{cases} 0 \text{ if } Max_\theta = Min_\theta \\ \frac{|i-j|}{|Max_\theta - Min_\theta|} \text{ otherwise} \end{cases} \tag{7}$$

$$\Delta_{Symb} = \begin{cases} 0 \text{ if } i = j \\ 1 \text{ otherwise} \end{cases} \tag{8}$$

The clustering error E_c measures the difference between the obtained partition and the real one in term of misclassification. It can be formalised as follows by

Datas	# Clusters Found		%Clustering Error	
	10_M [σ_{cf}]	A_C [σ_{cf}]	10_M [σ_e]	A_C [σ_e]
Art1	8.58 [0.98]	4.00 [0.00]	0.18 [0.01]	0.18 [0.02]
Art2	8.52 [0.96]	2.00 [0.00]	0.38 [0.01]	0.06 [0.02]
Art3	8.28 [0.96]	2.00 [0.00]	0.31 [0.01]	0.15 [0.02]
Art4	6.38 [0.75]	3.46 [0.50]	0.32 [0.02]	0.24 [0.05]
Art5	8.82 [0.91]	3.28 [0.45]	0.08 [0.01]	0.28 [0.03]
Art6	8.46 [1.08]	4.00 [0.00]	0.10 [0.02]	0.04 [0.01]
Art7	7.76 [1.03]	3.28 [0.45]	0.87 [0.02]	0.66 [0.02]
Art8	8.78 [0.83]	3.78 [0.42]	0.88 [0.01]	0.72 [0.04]
Iris	7.12 [1.11]	2.16 [0.37]	0.18 [0.03]	0.22 [0.01]
Glass	9.44 [0.70]	3.62 [0.64]	0.29 [0.02]	0.39 [0.03]
Pima	9.90 [0.36]	2.66 [0.56]	0.50 [0.01]	0.45 [0.01]
Soybean	8.82 [0.97]	4.42 [0.57]	0.13 [0.02]	0.07 [0.04]
Thyroid	9.56 [0.57]	2.88 [0.33]	0.42 [0.02]	0.18 [0.06]

Table 2: Results obtained after 50 runs of each method applied over each data.

considering all pairs of objects:

$$E_c = \frac{2}{N(N-1)} \times \sum_{(i,j)\in\{1,\ldots,N\}^2, i<j} \epsilon_{ij} \tag{9}$$

where:

$$\epsilon_{ij} = \begin{cases} 0 \text{ if } (c(o_i) = c(o_j) \wedge c'(o_i) = c'(o_j)) \vee \\ (c(o_i) \neq c(o_j) \wedge c'(o_i) \neq c'(o_j)) \\ 1 \text{ else} \end{cases} \tag{10}$$

with $c(o)$ the expected cluster identifiant for object o in the original data and $c'(o)$ the cluster found by the evaluated algorithm.

4.3 Results

The table 2 shows the number of clusters effectively found by both methods (*"#Clusters Found"*) with the standard deviation (*"*σ_{cf}*"*) and finally the error generated by both algorithms (*"%Clustering Error"*) associated with its standard deviation too(*"*σ_e*"*).

Our algorithm ANTCLUST performs better than the 10-MEANS method. It seems to be mainly because ANTCLUST manages to have, in general, a better appreciation of the number of clusters in the data. 10-MEANS finds too many

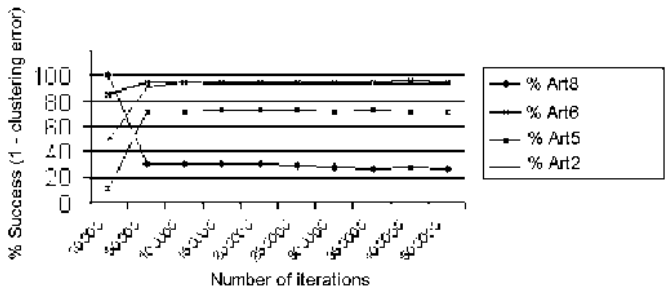

Figure 3: Success of ANTCLUST for several number of iterations. Success is equal to $1-$ clustering error.

clusters because it starts from 10 and it does not manage to reduce this number because of the too little difference existing between the objects. In fact, 10-MEANS performs better than ANTCLUST only twice: for ART5 and GLASS, because the number of clusters expected is quite near 10. These results show that ANTCLUST can treat small to big sets of data with a great success (see SOYBEAN, ART1, ART2 and ART6) but also demonstrate, that ANTCLUST does not manage to find a good partition when an important number of clusters is expected (see ART5 for instance). This may be due to the fact that there is only one rule that can create a new nest. This rule mainly applies at the beginning of the algorithm because even when an ant is ejected from its nest, it often remains alone and thus can not create a new nest, as two ants are needed.

We study the influence of the number of iteration to see if ANTCLUST could be enhanced. The number of iteration defines the number of possible meetings between the ants so it is a very restrictive parameter. Some tests allow us to verify that the number of iterations, that was inititially set to 300000, could be set to 100000 with similar results as shown in figure 3. With 50000 iterations or less, our algorithm does not manage to maintain the quality of the results because there is a large variability in clustering error and in the number of clusters found, especially for ART1, ART2 and ART5. In fact, when the number of iterations is no sufficient ANTCLUST generates too many clusters: this may be because the ants do not have enough time to gather correctly.

5 Conclusion

In this paper we describe a new model of the ant recognition system and its first application to the unsupervised clustering problem. Results are good when compared

to those of the 10-MEANS algorithm. Our approach does not make any assumption about the nature of the data to be clustered and does not require an initial partition or an initial number of classes. This allows us to test our method in numerous application fields. The first one will be the web mining problem and more precisely the study of the behaviour of Internet users, because of the growing necessity for such tools for webmasters and because it provides a huge source of data.

We are currently working on a new version of ANTCLUST that allows the user to see the generation of the nests in a 2D-space in real time. This version will rely on an other modelling of the Label and its evolution which will be more accurate. In fact, there are numerous ways left to adapt the mathematical model of the ants recognition system to the unsupervised clustering problem.

References

[1] R Boulay and A. Lenoir, 'Social isolation of mature workers affects nestmate recognition in the ant *camponotus fellah*', *Behavioral Processes*, **55**, 67–73, (2001).

[2] N.F. Carlin and B. Hölldobler, 'The kin recognition system of carpenter ants(camponotus spp.). i. hierarchical cues in small colonies', *Behav Ecol Sociobiol*, **19**, 123–134, (1986).

[3] D. Cliff, P. Husbands, J.A. Meyer, and Stewart W., eds. *Third International Conference on Simulation of Adaptive Behavior: From Animals to Animats 3*. MIT Press, Cambridge, Massachusetts, 1994.

[4] A. Colorni, M. Dorigo, and V. Maniezzo, 'Distributed optimization by ant colonies', in *Proceedings of the First European Conference on Artificial Life*, eds., F. Varela and P. Bourgine, pp. 134–142. MIT Press, Cambridge, Massachusetts, (1991).

[5] D. Fresneau and C. Errard, 'L'identité coloniale et sa reprsentation chez les fourmis', *Intellectica*, **2**, 91–115, (1994).

[6] B. Hölldobler and E.O. Wilson, *The Ants*, chapter Colony odor and kin recognition, 197–208, Springer Verlag, Berlin, Germany, 1990.

[7] A. K. Jain and Dubes R.C., *Algorithms for clustering Data*, chapter Square-Error Clustering Method, 96–101, Prentice Hall Advanced Reference series, 1988.

[8] P. Kuntz and D. Snyers, 'Emergent colonization and graph partitioning', In Cliff et al. [3], pp. 494–500.

[9] N. Labroche, N. Monmarché, A. Lenoir, and G. Venturini, 'Modélisation du système de reconnaissance chimique des fourmis', Rapport interne, Laboratoire d'Informatique de l'Université de Tours, E3i Tours, (january 2002). 45 pages.

[10] S. Lahav, V. Soroker, R.K. Vander Meer, and A. Hefetz, 'Nestmate recognition in the ant *cataglyphis niger*: Do queens matter?', *Behav Ecol Sociobiol*, **43**, 203–212, (1998).

[11] A. Lenoir, D. Cuisset, and A. Hefetz, 'Effects of social isolation on hydrocarbon pattern and nestmate recognition in the ant *aphaenogaster senilis*', *Insectes soc.*, **48**, 101–109, (2001).

[12] E.D. Lumer and B. Faieta, 'Diversity and adaptation in populations of clustering ants', In Cliff et al. [3], pp. 501–508.

[13] N. Monmarché, M. Slimane, and G. Venturini, 'On improving clustering in numerical databases with artificial ants', in *Lecture Notes in Artificial Intelligence*, eds., D. Floreano, J.D. Nicoud, and F. Mondala, pp. 626–635, Swiss Federal Institute of Technology, Lausanne, Switzerland, (13-17 September 1999). Springer-Verlag.